

Patrick Pataky

📞 +41 79 838 43 49 | ✉ patrick.pataky@epfl.ch | 📍 Renens, Switzerland | [in](#) ppataky | [🌐](#) Website



Master's student with a **strong focus on FPGA and hardware/software co-design**. Experienced in the full embedded design lifecycle, from **profiling software in C/C++** to **building cycle-accurate simulators** and **engineering custom RTL accelerators in Verilog**.

EDUCATION

- **Ecole polytechnique fédérale de Lausanne (EPFL)** 2024 (In Progress)
Master in Computer Science (with Cyber Security Specialization)
- **Ecole polytechnique fédérale de Lausanne (EPFL)** 2021 - 2024
Bachelor in Computer Science

TECHNICAL SKILLS AND INTERESTS

Languages: SystemVerilog, Verilog, VHDL, C/C++, Python, CUDA, Java, Scala, Bash, ARM, RISC-V, x86

Tools & Frameworks: Vivado, QuestaSim, FPGA (Xilinx/Intel), Quartus, Yosys, Linux, Git, GDB, TCL Scripting

Areas of Interest: Computer Architecture, Hardware-Software Co-Design, RTL Design & Microarchitecture, Compilers, Performance Optimization

Languages Spoken: English (fluent), French (native), Slovak (basic)

PROJECTS

- **X-HEEP RISC-V MCU: FPGA Verification and Cache Design** ([GitHub](#)) Spring 2026 (In Progress)
 - **FPGA Validation of FLASH Controller:** Validated read/write paths of the SPI Flash Controller on Xilinx FPGA for the open-source X-HEEP 32-bit RISC-V MCU, via end-to-end functional verification.
 - **Verification in C and On-Chip Debug:** Developed C test applications for read/write/erase robustness; used Integrated Logic Analyzer (ILA) and QuestaSim to debug and document controller limitations.
 - **Cache Optimization in RTL:** Implemented a SystemVerilog write-back cache (1 to 4 sectors per line) with hit/miss detection and line replacement, reducing expensive off-chip SPI transactions for recurrent Flash accesses.
- **FPGA Snake Game Hardware Accelerator** ([GitHub](#)) Fall 2024
 - **Graphics Pipeline RTL Design:** Developed a custom hardware graphics accelerator on the Gecko 5 Education Board FPGA, offloading display rendering from the CPU with dedicated DMA-based pixel data movement.
 - **Tile Engine & Animation Control:** Implemented a tile-by-tile custom FSM renderer to compose each frame and drive per-frame pixel animations for game objects under real-time constraints.
 - **Performance Outcome:** Achieved a 50x speedup over the software version by profiling C code to identify bottlenecks and mapping the critical rendering path to optimized Verilog RTL.
- **Nintendo DS Multiplayer Game** ([GitLab](#)) Fall 2023
 - **System Architecture:** Developed a multiplayer "Overcooked"-style game on Nintendo DS using both ARM9/ARM7 processors, splitting gameplay/graphics tasks and platform I/O services.
 - **Graphics & Real-Time Rendering:** Implemented a dual-screen graphics pipeline with tiled and rotscale backgrounds, sprite-based animations, and tile-map updates under strict memory and frame-time constraints.
 - **Embedded Integration:** Added timer/interrupt-driven synchronization, keypad and touchscreen controls, WiFi-based multiplayer synchronization, audio effects, and high-score persistence in cartridge storage.

EXPERIENCE

- **Teaching Assistant** Fall 2023 (In Progress)
EPFL, Lausanne (Switzerland)
 - **Courses:** CS-173 (Fundamentals of Digital Systems), CS-200 (Computer Architecture), CS-302 (Parallelism and Concurrency in Software), CS-470 (Advanced Computer Architecture)
 - **Delivered presentations and co-created course material**, including assignments and exams.
 - **Mentored 40-50 students weekly** providing guidance on digital logic, processor design, and programming labs.
 - **Debugged student implementations in real-time** for key projects, including designing and testing a simple in-order multicycle CPU from scratch in Verilog and writing a cycle-accurate C simulator of an out-of-order CPU.
- **IT Developer / Infrastructure Developer** August 2023-September 2023
Abalance SA, Neuchâtel (Switzerland)
 - **Server installation:** Installed and configured Linux servers for a 10-person environment, and reduced manual setup time by 90% by automating repeatable steps with shell scripts.
 - **Maintenance:** Performed regular maintenance tasks to ensure updates across 3 servers.